# Evaluation of systems' availability using a decomposition method, based on identification of independent subnets and failure trajectories

Dumitru Cezar Ionescu
POLITEHNICA University of Bucharest, Romania
*dc_ionescu@rectorat.pub.ro*

Adrian Cornel Constantinescu
POLITEHNICA University of Bucharest, Romania
*adrianconst@cne.pub.ro*

Anatoli Paul Ulmeanu
POLITEHNICA University of Bucharest, Romania
*paul@fiab.pub.ro*

## 1. Introduction

The modelling of complex systems is a very difficult task especial because of big number of components and dependencies between them. The Petri net represents one of the most attractive formalisms used to analyse such systems, because it may model any type of structural dependency and several types of time dependency as well. Anyway, the Petri is hard to evaluate in case of complex systems with many components. For building and analysing a Petri net two main techniques are available. Composition methods use subnet modules, which are combined in order to obtain the final net. Decomposition methods are based on splitting the net into subnets, which are evaluated separately, the obtained results being used to evaluate the whole net *(Tuffin 2000)*. In some cases, the subnets may communicate each other through tokens *(Gustavson 1995)*.

## 2. Petri net decomposition method

The present method can be used for timed Petri nets as GSPN or ESPN. In order to evaluate the systems' availability, a special part of the net is developed for checking whether the current state of the system is successfully or not. This subnet might be very useful especially for complex systems, which logical expression of success states is too difficult to be developed. In such cases the Petri net can be built as follows:
a) timed subnets modelling the systems' components and their functional and stochastic dependencies
b) a quotient net, that contains only immediate transitions, used to determine whether the current state of the system is successful or not *(Ionescu 2002)*.
In order to evaluate the systems' availability in a much easier manner, we have to take advantage of subsystems' independence and to do that we have to use a decomposition method. The proposed method is based on identification of subnets that contains at least a timed transition and that are independent each other. This means that anything happens inside a subnet nothing changes in the others except the quotient net. Before presenting the decomposition method, we introduce the following definitions.
**Definition 1.** The dependency relations between two nodes of the net (places/transitions) are defined by:

a) A transition *t* depends directly by a place *p* ( $p \overset{D}{\Rightarrow} t$ ) iff there is an input arc (normal or inhibitor) from *p* to *t* or if *p* appears in *t*'s predicate.

b) A place *p* depends directly by a transition *t* ( $t \overset{D}{\Rightarrow} p$ ) iff there is a normal arc from *t* to *p*.

c) Node $n_1$ depends indirectly by node $n_2$ ( $n_2 \overset{D}{\Rightarrow} n_1$ ) iff there is chain of direct dependencies from $n_2$ to $n_1$.

**Definition 2.** An independent subnet is a part of the Petri net that has no dependency relations with any outer timed transitions and that meets the following requirements:
a) The subnet contains at least one timed transition.

b) It includes only timed transitions *t* for that, there is at least one node *n* in the subnet such that $n \overset{D}{\Rightarrow} t$ .

c) It contains any node *n* for that, there is at least a timed transition *t* in the subnet such that $n \overset{D}{\Rightarrow} t$.

d) The subnet includes any node $n_1$ that has the following property: $\forall n_2, n_2 \overset{D}{\Rightarrow} n_1$, $n_2$ is included in the subnet.

e) The subnet includes only the arcs that connect nodes from it.

f) It contains no other independent subnet.

Definition 2 assures that, for a particular Petri net, there is only one way to divide it into independent subnets.

According to these definitions, in order to identify the independent subnets an algorithm was developed. It uses a recursive procedure (CHECK). CHECK procedure starts from a transition $t_0$. That represents the first component of the current subnet. All places *p* with property $p \overset{D}{\Rightarrow} t_0$ are added to the current subnet. If some of these places belong to other subnets too, these subnets and the current one are concatenated. For each of these places that were not checked before, there are identified the transitions *t* with property $t \overset{D}{\Rightarrow} p$. If some of these transitions belong to other subnets too, these subnets and the current one are concatenated. For each of these transitions that were not checked before, CHECK procedure is called. The recursive procedure ends when no more unchecked transitions are found.

The algorithm consists in two loops. In first one, each timed transition is checked, one by one. If the transition doesn't belong to any subnet, a new subnet is created for it and CHECK procedure is called. After that, in the second loop each node, which doesn't belong to any subnet, is checked whether it verifies point d of definition 2. If it does it is included in the corresponding subnet.

Following this algorithm all independent subnets and quotient net are identified.

## 3. Probabilistic evaluation method

Based on previous decomposition method, a probabilistic approach was developed in order to assess the system's availability. Before presenting the proposed approach, the following definitions are given.

**Definition 3.** A timed transition is called failure transition if its firing causes the unavailability of at least one system's component.

**Definition 4.** A failure marking (FM) is called minimal (MFM) if there is no other FM such that it is possible the net to pass from FM directly to MFM, following the firing of a failure transition.

So, a MFM is a FM with a minimal number of failures. By firing a failure transition, it's possible to arrive on MFM only by leaving a success marking (SM). Of course, there is possible to pass directly from a FM to MFM, but by firing a transition that is not a failure one. If the system, which is modelled with the Petri net, has binary and independent components, than a MFM represents a minimal cut-set of the system.

**Definition 5.** A failure trajectory (FT) is a trajectory during that the Petri net passes from initial marking (SM), through several intermediary markings (SM), to the final marking (FM).

Such a trajectory has the significance of an ordered set of timed transitions, the last one moving the net to a FM (see figure 1). Markings $M_0$, $M_1$, .., $M_{k-1}$ are SM but the final marking $M_k$ is a FM.

$$M_0 \xrightarrow{t_1} M_1 \xrightarrow{t_2} M_2 \xrightarrow{t_3} .. \xrightarrow{t_k} M_k$$

Figure 1: Failure trajectory

**Definition 6.** A minimal failure trajectory (MFT) is a FT that has the following property: there is no possible order of transitions passing such that the net reaches a FM, before the same final marking.

**Definition 7.** The marking $m_i^1$ of subnet *"i"* has a lower level of failure than marking $m_i^2$ (of the same

subnet), if there is at least one combination of markings $m_j, \forall j \neq i, 1 \leq j \leq N_S$ ($N_S$ is the number of independent subnets) for that the marking of the net $M\left(m_i^1, m_j\right)$ is a successful one and the marking $M\left(m_i^2, m_j\right)$ is a failure one. Otherwise, $m_i^1$ has a level of failure higher or equal to $m_i^2$.

**Definition 8.** The set of markings $M_{i,*}^k$, corresponding to a subnet *"i"* and a MFM *"k"*, is defined by: $\forall m_i \in M_{i,*}^k$, $m_i$ represents a level of failure higher or equal to $m_{i,*}^k$, where $m_i$ is a marking of subnet *i* and $m_{i,*}^k$ is the marking of subnet *"i"* corresponding to MFM *"k"*.

The probabilistic evaluation approach assumes the system modelled with Petri net has two characteristics:
a) It is coherent.
b) Each MFM can be reached by starting from $M_0$ and by firing only failure transitions.
According to previous definitions and hypotheses, the following theorems can be demonstrated.

**Theorem 1.** For any MFM, there is at least one MFT that links $M_0$ to it.

**Theorem 2.** The set $\bigcup_{k=1}^{N_F}\left(\bigcap_{i=1}^{N_S} M_{i,*}^k\right)$ represents the set of all FM ($N_F$ is the number of MFMs).

The first step, after net's decomposition and development of reachability graph for each independent subnet, is to find MFMs. In order to find all MFMs, the net has to be simulated and MFTs have to be found. According to theorem 1, identification of MFTs is sufficient to determine all MFMs. After MFMs identification, the next step is to find markings' sets $M_{i,*}^k$ given by definition 8. According to theorem 2, these sets can be used to express the set of failure markings *F*.

$$F = \bigcup_{k=1}^{N_F}\left(\bigcap_{i=1}^{N_S} M_{i,*}^k\right) \tag{1}$$

Therefore, by applying Sylvester-Poincare formula on equation (1), it follows:

$$P(F) = P\left(\bigcup_{k=1}^{N_F}\left(\bigcap_{i=1}^{N_S} M_{i,*}^k\right)\right) = \sum_{k=1}^{N_F}\left(\prod_{i=1}^{N_S} P\left(M_{i,*}^k\right)\right) - \sum_{k<j}^{N_F}\sum^{N_F-1}\left(\prod_{i=1}^{N_S} P\left(M_{i,*}^j \cap M_{i,*}^k\right)\right) + \sum_{k<j<n}^{N_F}\sum^{N_F-1}\sum^{N_F-2}\left(\prod_{i=1}^{N_S} P\left(M_{i,*}^j \cap M_{i,*}^k \cap M_{i,*}^n\right)\right) - \ldots \tag{2}$$

where $P\left(M_{i,*}^k\right)$ is the probability that subnet *"i"* to be in one of the markings included in $M_{i,*}^k$. As a result, we can calculate the probability the system to be in a failure state if we know the probability $P\left(m_i\right)$ the net to be in a marking $m_i$ of independent subnet *"i"*, $\forall i, 1 \leq i \leq N_S$. Probabilities $P\left(m_i\right)$ can be evaluated for each subnet separately, starting from any initial marking we want.

## 4. Study case

In order to test the method, a benchmark *(Bouissou 1998)* was solved using our dedicated software PNS *(Ionescu 2003)*, in which the present method was implemented. The analysed system includes the following components: one A component, two C components, two D components and eight E components. Each component has two possible states: up and down, and has different functional capacities (see figure 2.a). Therefore, the system may function at different capacity levels (100%, 90%, 70%, 60%, 40% and 30%). In order to evaluate the system's availability, a Petri net was developed (see

figure 2.b). Each immediate transition from the net was conditioned by a logical expression that is a function of places' tokens number (see table 1). PNS split the net into 4 independent subnets and a quotient net (see table 2). The possible markings of each subnet are given in table 3. For each capacity level MFMs were found and system's availability was evaluated (in table 4 there are presented MFMs and $M_{i,*}^{k}$ sets corresponding to first capacity level – 100%). The steady state availabilities calculated with PNS (using the new method and classical Markov process) and the ones provided by developer of the benchmark *(Bouissou 2000)* are presented in table 5. The absolute truncation error used for Sylvester Poincare formula was 1e-10.

Table 1: Logical expressions of immediate transitions

| | |
|---|---|
| T9: A=1 | T12: MIN(P9,MIN(4,4*C)+3*D)-P10<0 |
| T10: A=0 | T13: MIN(P10,1.5*E)-SUCCESS>0 AND E>5 |
| T11: MIN(P9,MIN(4,4*C)+3*D)-P10>0 | T14: MIN(P10,1.5*E)-SUCCESS<0 OR E<6 |

Table 2: Independent subnets

| | Subnet 1 | Subnet 2 | Subnet 3 | Subnet 4 | Quotient net |
|---|---|---|---|---|---|
| Places | A, P2 | C, P4 | D, P6 | E, P8 | P9, P10, SUCCESS |
| Transitions | T1, T2 | T3, T4 | T5, T6 | T7, T8 | T9, T10, T11, T12, T13, T14 |

Table 3: Subnet's markings

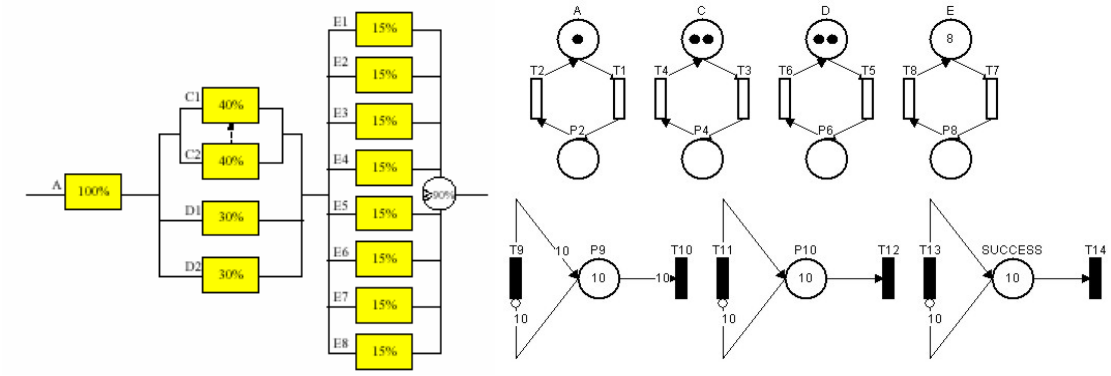
| Subnet 1 | Subnet 2 | Subnet 3 | Subnet 4 |
|---|---|---|---|
| (1,0); (0,1) | (2,0); (1,1); (0,2) | (2,0); (1,1); (0,2) | (8,0); (7,1); (6,2); (5,3); (4,4); (3,5); (2,6); (1,7); (0,8) |

Table 4: MFMs and $M_{i,*}^{k}$ sets – 100% capacity

| MFM 1 | $M_{i,*}^{1}$ sets | MFM 2 | $M_{i,*}^{2}$ sets |
|---|---|---|---|
| $m_{1,*}^{1}$: (0,1) | $M_{1,*}^{1}$: (0,1) | $m_{1,*}^{2}$: (1,0) | $M_{1,*}^{2}$: (1,0); (0,1) |
| $m_{2,*}^{1}$: (2,0) | $M_{2,*}^{1}$: (2,0); (1,1); (0,2) | $m_{2,*}^{2}$: (0,2) | $M_{2,*}^{2}$: (0,2) |
| $m_{3,*}^{1}$: (2,0) | $M_{3,*}^{1}$: (2,0); (1,1); (0,2) | $m_{3,*}^{2}$: (2,0) | $M_{3,*}^{2}$: (2,0); (1,1); (0,2) |
| $m_{4,*}^{1}$: (8,0) | $M_{4,*}^{1}$: (8,0); (7,1); (6,2); (5,3); (4,4); (3,5); (2,6); (1,7); (0,8) | $m_{4,*}^{2}$: (8,0) | $M_{4,*}^{2}$: (8,0); (7,1); (6,2); (5,3); (4,4); (3,5); (2,6); (1,7); (0,8) |
| MFM 3 | $M_{i,*}^{3}$ sets | MFM 4 | $M_{i,*}^{4}$ sets |
| $m_{1,*}^{3}$: (1,0) | $M_{1,*}^{3}$: (1,0); (0,1) | $m_{1,*}^{4}$: (1,0) | $M_{1,*}^{4}$: (1,0); (0,1) |
| $m_{2,*}^{3}$: (2,0) | $M_{2,*}^{3}$: (2,0); (1,1); (0,2) | $m_{2,*}^{4}$: (2,0) | $M_{2,*}^{4}$: (2,0); (1,1); (0,2) |
| $m_{3,*}^{3}$: (1,1) | $M_{3,*}^{3}$: (1,1); (0,2) | $m_{3,*}^{4}$: (2,0) | $M_{3,*}^{4}$: (2,0); (1,1); (0,2) |
| $m_{4,*}^{3}$: (8,0) | $M_{4,*}^{3}$: (8,0); (7,1); (6,2); (5,3); (4,4); (3,5); (2,6); (1,7); (0,8) | $m_{4,*}^{4}$: (6,2) | $M_{4,*}^{4}$: (6,2); (5,3); (4,4); (3,5); (2,6); (1,7); (0,8) |

Table 5: tem's availabilities

| | 100% | 90% | 70% | 60% | 40% | 30% |
|---|---|---|---|---|---|---|
| Proposed method | 9.6552E-01 | 9.3223E-03 | 1.9497E-02 | 1.1605E-03 | 9.7485E-05 | 2.3211E-05 |
| Markov process | 9.6552E-01 | 9.3223E-03 | 1.9497E-02 | 1.1605E-03 | 9.7485E-05 | 2.3211E-05 |
| (Bouissou 2000) | 9.6552E-01 | 9.3223E-03 | 1.9497E-02 | 1.1605E-03 | 9.7485E-05 | 2.3211E-05 |




a. b.

Figure 2: a. Capacity diagram of the system; b. The Petri net

## 5. Conclusions

In this paper a new method for evaluation of systems' availability was presented. This new method has some important advantages. It allows the identification and hierarchy of system's failure states based on levels of failure at independent subnets' level. Also, in order to evaluate the probabilities of subnets' markings, different probabilistic evaluation methods can be used (Markov process, Monte Carlo simulation etc.) depending the nature of each subnet's transitions. For this reasons, in case of multi-components systems, the computing time required by this method is much smaller than the one of a classical approach.

## References

M. Bouissou (1998) – "Miniplant: a benchmark for availability, reliability and performance evaluation", EDF, Direction des études et recherches

M. Bouissou (2000) – "Deux méthodes originales pour calculer les performances d'un système possédant des états de fonctionnement dégrade", EDF, Div. R&D, Department ESF

D.C. Ionescu, E. Zio, A.C. Constantinescu (2002) – "Efficient computation techniques for power systems reliability assessment" – "Mathematical and Statistical Methods in Reliability", published by World Scientific Publishing Co. in the series "Quality, Reliability and Engineering Statistics", p. 251-267

D.C. Ionescu, E. Zio, A.C. Constantinescu (2003) – "Availability analysis of a safety system of a nuclear reactor by means of a Generalized Stochastic Petri Net and forced Monte Carlo simulation" – Proceedings of "The 3rd Safety and Reliability International Conference Konbin", vol. 2, p. 225-232, Poland

Å. Gustavson, A. Törn, Å. Akademi (1995) – "Decomposing simulation nets by token communication" – Proceedings of the Conference on Modeling and Simulation ESM 95, Prague

B. Tuffin and K. S. Trivedi (2000) – "Implementation of importance splitting techniques in stochastic Petri net package" – in Computer Performance Evaluation: Modeling Techniques and Tools: Proceedings of the 11th International Conference, TOOLS 2000, Schaumburg, IL, p. 216-229